

## BAB II

### TINJAUAN PUSTAKA DAN DASAR TEORI

#### 2.1 Tinjauan Pustaka

Tinjauan pustaka dalam penelitian ini merupakan referensi penulisan dalam membangun perangkat sarung tangan yang digunakan untuk melakukan Konversi Huruf Alfabet dan Angka Dalam Audio Berbasis Arduino. Referensi penelitian ditunjukkan pada tabel 2.1

**Tabel 2.1 Tinjauan Pustaka**

No	Nama Peneliti	Perangkat pendukung	Interface	Hasil
1	Cici Ardianti, Azhar Al Havis, Munawir (2018)	<i>Webcam</i> laptop	Menggunakan layar pada laptop	Menampilkan informasi hasil <i>hand recognition</i> bahasa isyarat angka 1-10
2	Muktiaji Rofiandaru (2013)	Komputer	Form berisi gambar dan video	Menampilkan peragaan bahasa isyarat SIBI dengan gambar dan video
3	Zhulfi Bajra Wikjatmiko (2014)	<i>Smartphone android</i>	Menggunakan aplikasi <i>smartphone</i> dengan form berisi gambar dan video	Menampilkan peragaan bahasa isyarat dengan gambar dan video
4	Agung W. Setiawan, Lavita N. Rizalputri, Ahmad H.Thias (2018)	Sensor <i>flex</i> dan modul 3-axis <i>gyroscope</i> MPU6050	Menggunakan layar pada laptop	Menampilkan teks hasil gerakan jari dan posisi tangan
5	Anggun Mustikasari Dewi, Angga Rusdinar, Porman Pangaribuan (2018)	<i>Handphone</i> , Sensor <i>flex</i> , modul MPU6050, modul <i>Bluetooth</i> , dan <i>wemos shield</i>	Aplikasi <i>handphone</i>	Menampilkan hasil terjemahan bahasa isyarat berupa teks dan suara

No	Nama Peneliti	Perangkat pendukung	Interface	Hasil
6	Usulan Muhammad Firdaus Nurrohim	Sensor <i>flex</i> , modul <i>gyroscope- accelerometer</i> , modul <i>micro SD</i> , dan <i>speaker</i>	Suara pada <i>speaker</i>	Menampilkan hasil terjemahan Bahasa isyarat SIBI berupa suara huruf a-z dan angka 1-10

Pada jurnal nomor 1 yang ditulis oleh Cici Ardianti, Azhar Al Havis, Munawir (2018) melakukan penelitian mengenai pengenalan gestur menggunakan webcam yang dapat mengenali gerak tangan isyarat manusia dan menterjemahkan gerakan tersebut sebagai instruksi yang dapat dipahami oleh komputer.

Pada jurnal nomor 2 yang ditulis oleh Muktiaji Rofiandaru (2013) menciptakan sistem pembelajaran Bahasa Isyarat SIBI dengan menggunakan pendekatan media foto dan/atau video.

Pada jurnal nomor 3 yang ditulis oleh Zhulfi Bajra Wikjatmiko (2014) melakukan penelitian mengenai aplikasi yang berisi video, suara, dan gambar yang berisi tata cara bahasa isyarat yang digunakan dalam proses komunikasi dengan tunarungu.

Pada jurnal nomor 4 yang ditulis oleh Agung W. Setiawan, Lavita N. Rizalputri, Ahmad H.Thias (2018) melakukan penelitian mengenai alat bantu komunikasi penderita pascastroke berbentuk sarung tangan dengan memanfaatkan gerakan jari dan tangan. Sarung tangan ini menggunakan sensor *flex* dan sensor *accelerometer*.

Pada jurnal nomor 5 yang ditulis oleh Anggun Mustikasari Dewi, Angga Rusdinar, Porman Pangaribuan (2018) melakukan penelitian mengenai alat bantu tunarungu berbentuk sarung tangan yang dapat mengenali bahasa isyarat. Sarung tangan ini menggunakan sensor *flex*, dan *bluetooth* yang digunakan untuk mengirimkan data ke aplikasi di *smartphone*.

Berdasarkan tinjauan pustaka di atas alat sarung tangan ini dibuat menggunakan sensor *flex* untuk lengkungan jari dan kombinasi sensor *gyroscope-accelerometer* untuk gerakan dan/atau kemiringan tangan yang berfungsi menerjemahkan gerakan bahasa isyarat SIBI berupa huruf alfabet a-z dan angka 1-10. Disamping itu perangkat *speaker* diletakkan pada sarung tangan sehingga proses konversi dari gestur ke suara bisa terjadi secara instan tanpa perangkat tambahan lainnya.

## **2.2 Dasar Teori**

Penelitian yang baik harus dilandasi dengan dasar teori mumpuni. Berdasarkan pemaparan pada latar belakang, dasar teori yang digunakan untuk penelitian ini yaitu :

### **2.2.1 Arduino**

Arduino adalah kit elektronik atau papan rangkaian elektronik bersifat *open source* yang di dalamnya terdapat komponen utama yaitu sebuah cip mikrokontroler dengan jenis *Advance Versatile RISC* (AVR) dari perusahaan Atmel. Cip ini berlaku sebagai otak dari Arduino, dengan bahasa pemrograman Arduino yaitu bahasa pemrograman yang mirip

dengan C/C++. Rangkaian elektronik dapat menerima input dan kemudian memprosesnya sehingga menjadi *output* yang diinginkan. Penerapan Arduino sangat sering digunakan dalam pembuatan robot, *remote control*, *running text* dan lain sebagainya. Arduino nano dan Arduino pada umumnya bekerja menggunakan pemrograman dengan bahasa Arduino yang dituliskan pada software Arduino IDE. Software Arduino IDE terdiri dari tiga bagian :

- a. Editor program, yaitu tempat untuk penulisan atau pengeditan program yang akan di tanamkan pada Arduino. Setiap program Arduino biasa disebut *sketch*.
- b. *Compiler*, yaitu modul yang berfungsi mengubah bahasa pemrograman kedalam kode biner, karena hanya kode biner yang dapat dipahami mikrokontroler.
- c. *Uploader*, yaitu modul yang berfungsi memasukan kode biner kedalam memori mikrokontroler.

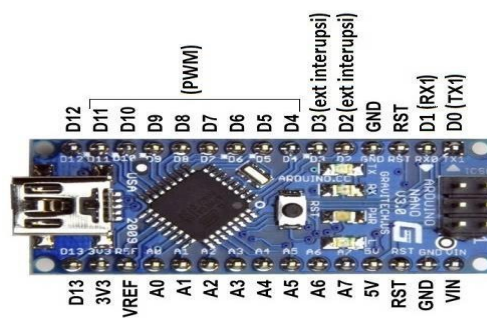
Arduino Nano adalah salah satu varian dari produk *board* mikrokontroller keluaran Arduino. Arduino Nano adalah *board* Arduino terkecil, menggunakan mikrokontroller Atmega 328 untuk Arduino Nano 3.x dan Atmega168 untuk Arduino Nano 2.x. Varian ini mempunyai rangkaian yang sama dengan jenis Arduino Duemilanove, tetapi dengan ukuran dan desain PCB yang berbeda. Arduino Nano tidak dilengkapi dengan soket catudaya, tetapi terdapat pin untuk catu daya luar atau dapat

menggunakan catu daya dari mini *USB port*. Arduino Nano didesain dan diproduksi oleh Gravitech.

Arduino Nano mempunyai 14 pin digital yang dapat digunakan sebagai pin input atau *output*. Pin ini akan mengeluarkan tegangan 5V untuk mode *HIGH* (logika 1) dan 0V untuk mode *LOW* (logika 0) jika dikonfigurasi sebagai pin *output*. Jika di konfigurasi sebagai pin input, maka ke 14 pin ini dapat menerima tegangan 5V untuk mode *HIGH* (logika1) dan 0V untuk mode *LOW* (logika 0). Besar arus listrik yang diijinkan untuk melewati pin digital I/O adalah 40 mA. Pin digital I/O ini juga sudah dilengkapi dengan resistor pull-up sebesar 20-50 k $\Omega$ . Ke 14 pin digital I/O ini selain berfungsi sebagai pin I/O juga mempunyai fungsi khusus yaitu : pin D0 dan pin D1 juga berfungsi sebagai pin TX dan RX untuk komunikasi data serial. Kedua pin ini terhubung langsung ke pin IC FTDI USB-TTL. Pin D2 dan pin D3 juga berfungsi sebagai pin untuk interupsi eksternal. Kedua pin ini dapat dikonfigurasi untuk pemicu interupsi dari sumber eksternal. Interupsi dapat terjadi ketika timbul kenaikan atau penurunan tegangan pada pin D2 atau pin D3. Pin D4, pin D5, pin D6, pin D9, pin D10 dan pin D11 dapat digunakan sebagai pin *Pulse Width Modulator* (PWM). Pin D10, pin D11, pin D12 dan pin D13, ke empat pin ini dapat digunakan untuk komunikasi mode SPI. Pin D13 terhubung ke sebuah LED.

Arduino Nano juga dilengkapi dengan 8 buah pin analog, yaitu pin A0, A1, A2, A3, A4, A5, A6 dan A7. Pin analog ini terhubung ke Analog

to Digital Converter (ADC) internal yang terdapat di dalam mikrokontroller. Pada kondisi awal, pin analog ini dapat mengukur variasi tegangan dari 0 V sampai 5 V pada arus searah dengan besar arus maksimum 40 mA. Lebar *range* ini dapat diubah dengan memberikan sebuah tegangan referensi dari luar melalui pin Vref. Pin analog selain dapat digunakan untuk input data analog, juga dapat digunakan sebagai pin digital I/O, kecuali pin A6 dan A7- yang hanya dapat digunakan untuk input data analog saja. Fungsi khusus untuk pin analog antara lain : Pin A4 untuk pin SDA, pin A5 untuk pin SCL, pin ini dapat digunakan untuk komunikasi I2C. Pin Aref digunakan sebagai pin tegangan referensi dari luar untuk mengubah range ADC. Pin reset, pin ini digunakan untuk mereset *board* Arduino Nano, yaitu dengan menghubungkan pin ini ke *ground* selama beberapa milidetik. *Board* Arduino Nano selain dapat direset melalui pin reset, juga dapat direset dengan menggunakan tombol reset yang terpasang pada *board* Arduino Nano.



**Gambar 2.1 Arduino Nano**

### 2.2.2 Sensor *Flex*

Sensor *Flex* adalah sensor yang nilai hambatannya bergantung pada tekukan yang terjadi di sensor ini. Dalam hal ini, hambatan akan membesar apabila sensor ini dibengkokkan.



**Gambar 2.2 Sensor Flex**

### 2.2.3 Sensor MPU-6050 (Gyroscope-Accelerometer)

*Gyroscope* adalah peranti yang dapat mengindera objek yang berputar, sedangkan *accelerometer* adalah peranti yang mengukur gerakan objek. Pada *accelerometer* menghasilkan percepatan pada tiga sumbu, sedangkan *gyroscope* menghasilkan kecepatan putar pada tiga sumbu (dengan satuan derajat per detik). IC MPU-6050 mengombinasikan fungsi *gyroscope* dan *accelerometer*. Oleh karena itu, IC ini mampu mendeteksi putaran maupun gerakan dalam tiga sumbu (yaitu Y, P, dan R).



**Gambar 2.3 Sensor MPU-6050**

### 2.2.4 Modul MicroSD Card

Modul *MicroSD* merupakan modul untuk mengakses *Micro SD* untuk pembacaan maupun penulisan data dengan menggunakan sistem antarmuka *Serial Parallel Interface* (SPI).

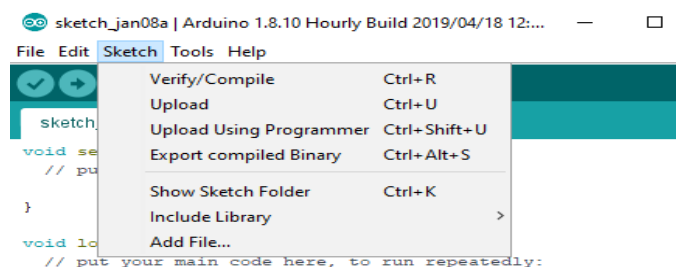


**Gambar 2.4 Modul *MicroSD Card***

Modul ini cocok untuk berbagai aplikasi yang membutuhkan media penyimpanan data, seperti sistem absensi, sistem antrian, maupun sistem aplikasi data logging lainnya.

### 2.2.5 *Library*

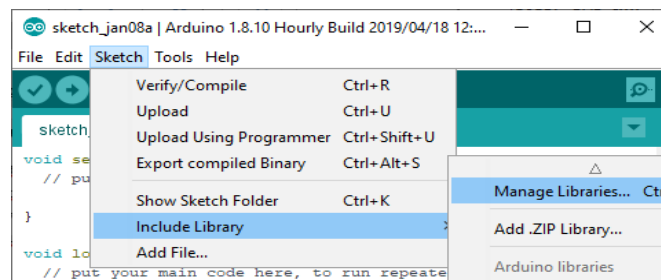
Lingkungan Arduino dapat diperluas melalui penggunaan *Library*, seperti kebanyakan platform pemrograman. *Library* menyediakan fungsionalitas tambahan untuk digunakan dalam sketsa, misal bekerja dengan perangkat keras atau memanipulasi data. Untuk menggunakan *library* dalam sketsa, pilih dari menu *Sketch* kemudian *Include Library* kemudian pilih *Import Library*.



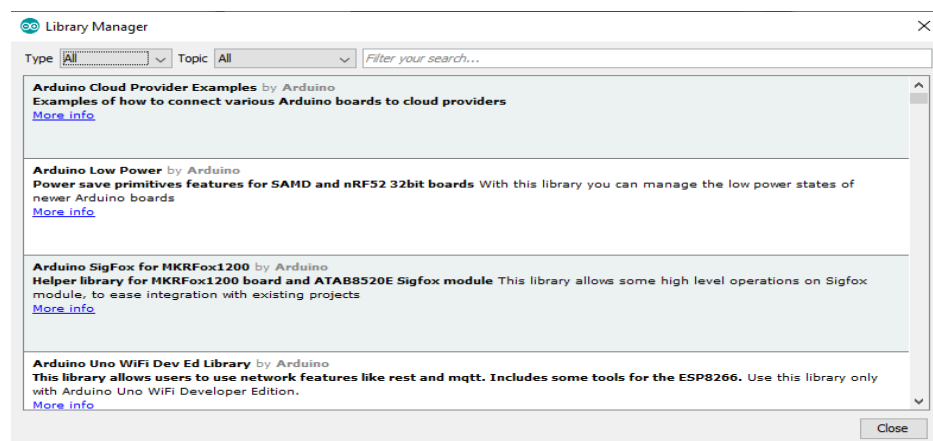
**Gambar 2.5 Menu *Sketch***

Sejumlah *library* terinstal dengan IDE, tetapi kita juga dapat mengunduh atau membuat milik kita sendiri. Ada juga tutorial menulis *library* kita sendiri. Lihat Panduan Gaya API untuk informasi tentang membuat API gaya Arduino yang bagus untuk *library* kita.





**Gambar 2.6 Manage Libraries**



**Gambar 2.7 Beberapa Library Arduino**

### 2.2.6 Metode Map

Metode *map* merupakan sebuah peta konsep yang bertugas untuk mengumpulkan semua data yang penting pada konsep program yang kemudian dikumpulkan dalam satu data. Metode *map* dilakukan dengan cara mengumpulkan semua nilai data kemudian memetakannya menjadi satu konsep.

Mapping adalah fungsi yang akan mengubah satu *range* data, menjadi *range* data yang baru. Sebagai contoh, input data dari Analog Input Potentiometer. Kita tahu, *range* data yang akan didapati, pastilah dari 0~1023. Dengan data tersebut, kita mungkin mahu mengawal kelajuan satu motor yang dipandu oleh Motor *Controller*. Kebanyakan Motor

*Controller* memerlukan input kawalan PWM, dan PWM *output* dari pin-pin PWM Arduino dapat dikawal dengan memberikan *value* dari 0~255 dalam koding. Dengan begitu, kita perlu mengubah *output* 0~1023 menjadi nilai 0~255 untuk mengawal Motor *Controller*. Berikut adalah cara dengan menggunakan formula matematika yang sedikit *complicated* :

```
dataSensor = analogRead(A1);
dataBaru = dataSensor / (1023/255);
// 1023/255 = 4.011
// dataBaru = 1023 / 4.011
// dataBaru = 255
```

Dengan formula di atas, anda dapat mengubah data 0~1023 tersebut menjadi data baru 0~255, namun setiap kali ada *range* baru yang ingin dibuat, perlu membuat kiraan formula sekali-lagi.

```
dataSensor = analogRead(A1);
dataBaru = map(dataSensor, 0, 1023, 0, 255);
```

Di atas merupakan koding mapping untuk menggantikan koding matematika di permulaan tadi. Algoritma yang digunakan adalah algoritma `map()` sehingga tidak perlu membuat kiraan saat ada *range* baru. Adapun sintaks dan fungsi `map` untuk metode *map* sebagai berikut :

```
dataBaru = map(x, in_min, in_max, out_min,
out_max);

map(x, in_min, in_max, out_min, out_max){
(x - in_min) * (out_max - out_min) / (in_max
- in_min) + out_min}
```